```
/*************   DIRECT DIAGONALIZATION                  ******************/

double a,b,c,d,  e,f,g,h,  i,j,k,l,  m,n,o,p; /* synomyms*/
a =  ATA_sum[0][0]; b =  ATA_sum[0][1]; c =  ATA_sum[0][2]; d =  ATA_sum[0][3];
e =  ATA_sum[1][0]; f =  ATA_sum[1][1]; g =  ATA_sum[1][2]; h =  ATA_sum[1][3];
i =  ATA_sum[2][0]; j =  ATA_sum[2][1]; k =  ATA_sum[2][2]; l =  ATA_sum[2][3];
m =  ATA_sum[3][0]; n =  ATA_sum[3][1]; o =  ATA_sum[3][2]; p =  ATA_sum[3][3];

double A, B, C, D, E; /* coefficients id the characteristic polynomial - according to sympy …
A = 1.0;/* will not be used below */
B = (-a - f - k - p);
C = (a*f + a*k + a*p - b*e - c*i - d*m + f*k + f*p - g*j - h*n + k*p - l*o);
D = (-a*f*k - a*f*p + a*g*j + a*h*n - a*k*p + a*l*o + b*e*k + b*e*p - b*g*i
    - b*h*m - c*e*j + c*f*i + c*i*p - c*l*m - d*e*n + d*f*m - d*i*o + d*k*m
    - f*k*p + f*l*o + g*j*p - g*l*n - h*j*o + h*k*n) ;
E =  a*f*k*p - a*f*l*o - a*g*j*p + a*g*l*n + a*h*j*o - a*h*k*n - b*e*k*p
   + b*e*l*o + b*g*i*p - b*g*l*m - b*h*i*o + b*h*k*m + c*e*j*p - c*e*l*n
   - c*f*i*p + c*f*l*m + c*h*i*n - c*h*j*m - d*e*j*o + d*e*k*n + d*f*i*o
   - d*f*k*m - d*g*i*n + d*g*j*m;

double alpha, beta, gamma; /* substitutions */
alpha = -3*B*B/8 + C;
beta  = B*B*B/8  - B*C/2 + D;
gamma = -3*B*B*B*B/256 + C*B*B/16 - B*D/4 + E;

if ( fabs(beta) < 1.e-4) {
   double blah = alpha*alpha-4*gamma;
   if ( blah < 0) {
       /* what should I do in this case? bail out*/
       *rmsd = -1;
       return 1;
   }
   double sblah = sqrt(blah);
   double blah2;
   blah2 = (-alpha+sblah)/2;
   if ( blah2 < 0) {
       *rmsd = -1;
       return 1;
   }
   w[0] = -B/4+sqrt(blah2);
   w[1] = -B/4-sqrt(blah2);

   blah2 = (-alpha-sblah)/2;
   if ( blah2 < 0) {
       *rmsd = -1;
       return 1;
   }
   w[2] = -B/4+sqrt(blah2);
   w[3] = -B/4-sqrt(blah2);
   /* we'll have to find the smallest lambda below */

} else { /* beta is significantly bigger than 0 */

   double P, Q, R, U, V, W;
   double blah;
   /* more substitutions */
   P = - alpha*alpha/12-gamma;
   Q = - alpha*alpha*alpha/108 - alpha*gamma/3 - beta*beta/8.0;
   blah = Q*Q/4+P*P*P/27;
   if ( blah< 0) {
       *rmsd = -1;
       return 1;
   }
   R = -Q/2+sqrt(blah);
   if ( fabs(R) <= 1.e-4 ){
       V = -5*alpha/6 - pow (Q, 0.333);
   } else {
       U = pow(R, 0.3333);
       V =  -5*alpha/6 + U -P/3/U;
   }

   if ( alpha + 2*V < 0 ){
```

```
       *rmsd = -1;
       return 1;
   }
   W = sqrt (alpha + 2*V);
   if ( fabs(W) < 1.e-10) {
       *rmsd = -1;
       return 1;
   }

   blah = -(3*alpha+2*V+2*beta/W);
   if ( blah < 0) {
       *rmsd = -1;
       return 1;
   }
   blah = sqrt(blah)/2;
   w[0] = -B/4 + W/2 - blah;
   w[1] = -B/4 + W/2 + blah;


   blah = -(3*alpha+2*V-2*beta/W);
   if ( blah < 0) {
       *rmsd = -1;
       return 1;
   }
   blah = sqrt(blah)/2;
   w[2] = -B/4 - W/2 - blah;
   w[3] = -B/4 - W/2 + blah;


}
```